PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 : H04N 7/173 — A1

(11) International Publication Number: WO 92/15174

(43) International Publication Date: 3 September 1992 (03.09.92)

(21) International Application Number: PCT/GB92/00337

(22) International Filing Date: 25 February 1992 (25.02.92)

(30) Priority data:
9103907.3 25 February 1991 (25.02.91) GB

(71) Applicant: BEAUMONT-MAXIN INTERNATIONAL LIMITED [GB/GB]; 66 Lincoln's Inn Fields, London WC2A 3LH (GB).

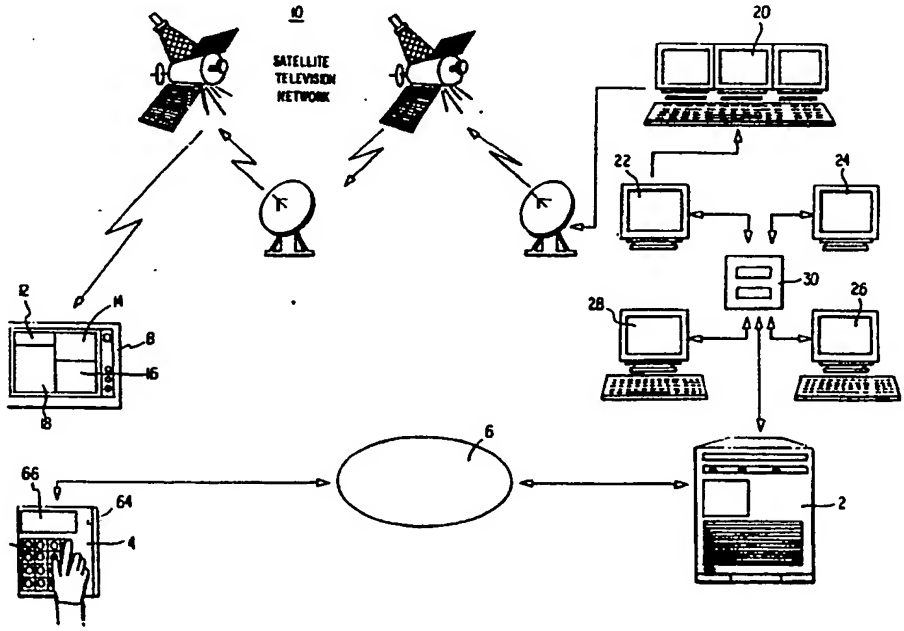
(72) Inventors: GRUSIN, Hillard, T. ; GRUSIN, Madeleine ; 2 Carlyle Court, Chelsea Harbour, London SW10 0UQ (US). THOMAS, Alexander ; Appleford, Station Road, Bentley, Farnham, Surrey GU10 5LA (GB). DIAMOND, Clifford, J. ; 13 Langley Crescent, Edgeware, Middlesex HA8 9SZ (GB). FERNANDES, Michael ; 22 Hathaway Close, Ruislip Gardens, Ruislip, Middlesex HA4 6PY (GB).

(74) Agents: CROUCH, David, John et al.; Bromhead & Co., 19 Buckingham Street, London WC2N 6EF (GB).

(81) Designated States: AT (European patent), BE (European patent), CH (European patent), DE (European patent), DK (European patent), ES (European patent), FR (European patent), GB (European patent), GR (European patent), IT (European patent), JP, LU (European patent), MC (European patent), NL (European patent), SE (European patent).

Published
With international search report.

(54) Title: INTERACTIVE TRANSACTION PROCESSING SYSTEM



(57) Abstract

A distributed object-based transaction system provides a plurality of terminals (4) and/or host computers (2) on which objects, or named memory spaces, reside. The objects are controlled by methods which are located on each respective terminal or host on which an object resides, and the methods can be invoked by any of the terminals or host computers. Updating the objects is accomplished by invoking the relevant method on each of the nodes wherein the object resides. The distributed object-based transaction system is particularly useful in the implementation of an auction system for remotely situated bidders utilizing interactive television.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FI	Finland	ML	Mali
AU	Australia	FR	France	MN	Mongolia
BB	Barbados	GA	Gabon	MR	Mauritania
BE	Belgium	GB	United Kingdom	MW	Malawi
BF	Burkina Faso	GN	Guinea	NL	Netherlands
BG	Bulgaria	GR	Greece	NO	Norway
BJ	Benin	HU	Hungary	PL	Poland
BR	Brazil	IE	Ireland	RO	Romania
CA	Canada	IT	Italy	RU	Russian Federation
CF	Central African Republic	JP	Japan	SD	Sudan
CG	Congo	KP	Democratic People's Republic of Korea	SE	Sweden
CH	Switzerland			SN	Senegal
CI	Côte d'Ivoire	KR	Republic of Korea	SU	Soviet Union
CM	Cameroon	LI	Liechtenstein	TD	Chad
CS	Czechoslovakia	LK	Sri Lanka	TG	Togo
DE	Germany	LU	Luxembourg	US	United States of America
DK	Denmark	MC	Monaco		
ES	Spain	MG	Madagascar		

INTERACTIVE TRANSACTION PROCESSING SYSTEM

TECHNICAL FIELD

This invention relates to the art of data processing and the combination of data processing with video displays of items related to the data. In the ultimate preferred embodiment, the invention is an auction system for remotely situated bidders conducted utilizing interactive television.

BACKGROUND

The invention is an interactive transaction processing system and data processing method. The fundamental system is a distributed object-based transaction system having a wide range of potential uses. This distributed object-based transaction system has particular utility in the implementation of an interactive televised auction in which remote subscribers bid in competition with the live auction in the saleroom. Thus, disclosure of the invention will be facilitated by first providing a basic appreciation of the operation of an auction.

An auction normally proceeds under the control of an auctioneer. An item to be auctioned (sold) is displayed, and the auctioneer asks for a "bid" for the item. The auctioneer can set an opening bid price. The auctioneer invites further bids without

specifying the next price. The auctioneer states the price when accepting the next bid, using an increment judged as reasonable given the number of bidders participating. Each of the bidders typically has a "paddle", having the bidder's number on it, which is held up to signal to the auctioneer that the bidder wishes to bid. The auctioneer usually accepts the bid of the first bidder to hold up his paddle and states the new price. The number on the paddle of the bidder whose bid has been accepted is recorded by the auctioneer or his clerk.

Saleroom etiquette dictates that if the auctioneer senses that two bidders are competing for the item being sold, he should conduct a "ping-pong" auction between these two bidders. A ping-pong auction is a process whereby the auctioneer ignores bids by bidders other than the chosen two bidders competing for the item. When one of the ping-pong bidders drops out, e.g., by failing to make a bid, the auctioneer will then accept bids from other bidders.

The system of the invention is designed to allow an auction in accordance with this process to be conducted at a plurality of remote sites. In general, terms used in the art of auctions will be used to describe the invention, and the bidders at the remote sites will be referred to as "subscribers". It will be appreciated, however, that the invention is equally applicable to a variety of operations other than auctions.

SUMMARY OF THE INVENTION

A. Distributed Object-based Data Processing

In accordance with the invention, a distributed object-based data processing program provides a system for handling a broad range of transactions and is specifically adapted to conduct an auction in the preferred embodiment. In accordance with the technique known as object-oriented design, the distributed object-based data processing program comprises "methods", "objects", and "classes".

An "object" is a data element, the specific definition of which depends upon the desired transaction or other operation. In the auction system which will be described in detail below, an example of an object is the "bid display." This is a list of identifications of the first ten bidders whose bids were received by the host computer in order of receipt of the bids.

A "method" is a set of instructions which are provided to the host computer to tell the computer what processes are to be performed on the object. For example, when a bid is received, the method "update bid display" is invoked which adds the identification of a bidder (another object) to the bid display or cancels a bid.

A "class" is a group of related methods, or routines. For example, the "user display class" includes the "add user" and "update user" methods. Classes can be arranged in a structure where one class "inherits" methods from another class (termed its

"superclass"). The ultimate "ancestor" class is the "root" class, which contains the most generally applicable methods. The methods are grouped into classes to maximize efficiency, minimize the impact of subsequent design changes and promote reusability of code. Also, the bulk of the source code is reduced since terms common to the grouped methods do not have to be redefined for each of the related methods.

The typical instruction to the computer, which can be programmed in any of a variety of languages is: "Perform method B on object A". This would cause the computer first to find the class of object A by searching a table of classes. Then, it searches the method table of that particular class to confirm that method B is indeed a part of that class. If the computer cannot find the particular method in the given class, it searches its superclass. The computer then invokes the particular method by calling a subroutine identified by the method and performing the steps which comprise the method.

After the particular method has been performed on the object, the system updates the value of that object at all nodes in which the particular object resides by invoking the same method at those nodes. Thus, if the object "bid display" resides on several workstations and the host, the new value of the "bid display" object resulting from the completion of an invoked method relating to that object will be propagated to all of the nodes by the update

process which invokes the same method at each of the remaining nodes having the "bid display" object.

The distributed object-based transaction system of the invention integrates an application which is divided into several processes and running on several machines. For example, an application may be terminal-based and intended for implementation on a single host computer, or it may be implemented on multiple workstations connected to a variety of other devices. In the auction example described below, personal computers have programs capable of implementing individual methods relating to the objects located on the particular personal computer. The invoking of a method, which inherently changes the value of an object, is always automatically communicated to the other workstations and the host computer having that object by the process of updating whereby that method in also invoked at all nodes having that object. Thus, the distributed object-based transaction system provides a uniform mechanism for interaction between application components while allowing each platform to contribute maximum functionality.

Remote procedure call mechanisms are known and give the programmer the ability to call a subroutine in the normal way but have it execute in another process that could be located in another machine. To the programmer, the result appears in exactly the same as if the subroutine had been part of his program and had been executed in the same process.

The remote procedure call mechanism used in the system of the invention, as well as that of other systems, such as Sun's RPC, employs a so-called 'stub' routine which takes the place of the real routine in the caller's code. The stub routine then communicates with the remote process and passes the necessary information to it so that the correct routine is invoked in the remote process. The remote process will then return any results of the subroutine call back to the stub routine, which will be waiting for the reply. The stub routine then unpacks the returned information and places it into the proper variables. The stub routine then returns control to the calling routine and execution continues as normal.

The distributed object-based transaction system of the invention implements a remote procedure call mechanism and routes requests and responses across external networks and internal inter-process communications structures (such as pipes and queues). The system can cope with multiple paths between two nodes and chooses the most efficient route available.

The distributed object-based transaction system allows the combination of an advanced, general purpose application architecture with specific support for host computer features.

By permitting any node (object location) to be either a client or a server, the distributed object-based transaction system transcends fixed client-server roles for networked computers. In the traditional system, a server offers a range of services and the

client accesses those services. In the distributed object-based transaction system of the invention, the server has a range of objects, and the client is one who accesses those objects. Any of the workstations may be a client with respect to some objects as well as a server with respect to other objects. This allows workstations, for instance, to be informed of dynamically changing information as well to initiate their own transactions.

The distributed object-based transaction system disclosed herein has the further advantage that all interactions between nodes on the network are based on a single model. The model follows the object-oriented paradigm and has the advantage of local transparency to the clients in that the client need only identify the data object in a call and not the servers associated with that object. For example, a read only request will be directed only to the server nearest the client, whereas an update would be propagated to all locations of the object.

A basic feature of the distributed object-based transaction system of the invention is that the data can be replicated because the objects can reside in multiple nodes. In the traditional system, the host is generally the server because it contains the database. In contrast, the distributed object-based transaction system allows each of the workstations to have a database related to the objects residing on that workstation. The system ensures consistency between multiple copies of the same data by routing update requests to all relevant servers for a given object. For

example, if a number of workstations display the value of a particular data object which also resides on the host, the system will invoke methods in each of the involved workstations to cause all of the values to be the same in all workstations concerned. There is, thus, no requirement to explicitly code for the distribution of the data because the system automatically updates the value of the object at all nodes wherein that object is located.

Transaction management, like shared-memory and concurrent processing support, is a specific feature of the Stratus machine, which is the preferred host computer. While equivalent facilities exist in a few other environments the distributed object-based system of the invention manages the Stratus version of them. Transaction protection ensures that every transaction will either complete successfully or be fully "backed-out", i.e., leave no trace that it ever executed. This feature is vital for transaction procession systems and is extremely complex to emulate if not available.

Without transaction protection, a database can become inconsistent if 1) two transactions interfere with each other, e.g. by updating the same record or 2) a transaction fails during execution leaving some updates performed but not others. A transaction protection system will ensure that two transactions do not interfere with each other and that, if one does not complete, it is fully backed-out.

The programmer normally has to call subroutines to indicate the start and end of transactions. Ending a transaction normally is called 'committing' because at this point the changes made take effect apparently instantaneously and cannot thereafter be undone. In the transaction, the programmer must check the status of every file operation to determine whether the transaction can proceed or whether the transaction protection system has detected a conflict. If a problem is detected, the programmer must abort the transaction, i.e. end it abnormally. Ideally, because conflicts can arise in the normal course of events, the programmer should attempt to execute the entire transaction again because the cause of the conflict (another transaction) will finish eventually.

Instead of the programmer having to code for these functions, the distributed object-based transaction system of the invention takes over the management of transactions. Because a transaction in accordance with the system of the invention corresponds to a method, the system will start the transaction before calling the method code and, when the method finishes, the system can commit. The system also has all the information at hand to restart transactions which fail. It does this by offering the programmer equivalents to all the file operation subroutines. These equivalents call the real file subroutine but check the status of the result. If they detect a conflict, the method is aborted at that point and control jumps back to the system which can call the transaction again automatically.

Another feature of the system of the invention is its port pool management. Before the data of a file can be accessed in a program, a port must be "opened" to that file. There is an "open" operating system routine which locates the file by name and creates a port through which that program can access it. Thereafter, the file is accessed by the given port number, not by its name. In the open call, the programmer specifies what kind of access is required, such as input or update, indexed or sequential. The port also remembers the program's current position in the file, so that calls can be made to read successive records or update the current record.

A batch program will open ports, perform file operations and then close them again. With an on-line, real-time system, ports cannot be opened for each transaction as this would create an unacceptable overhead. Instead, ports are opened once when the system starts and stay open while the system is up.

Because the system of the invention allows methods to be called from anywhere, including from other methods, a situation can arise where a method uses the same port as the method calling it. If, in using the port, the method altered the current position in the file then the calling method would lose its position. To avoid this type of interference the system ensures that methods called by other methods in the same process use different ports. It does this by maintaining a pool of ports which were opened when the system started. As many ports are opened as are likely to be

needed in the course of processing. Before a method can access a file it calls a system port allocation routine in the same way that it would have called the "open" routine. The port allocation routine finds a pre-opened port satisfying the type of access asked for and marks it as in use by this particular method. When the method completes, the system automatically marks the port free again (i.e., de-allocates it). If another method is called within the first method, then a different port will be allocated to it. The mapping of pooled ports to real ports is performed by the same substitute file operation subroutines that are responsible for detecting transaction protection conflicts outlined above.

B. Distributed Object-based Data Processing Applied to a Televised Auction System

Application of a distributed object-based data processing system to a televised auction system in accordance with a preferred embodiment requires the following objects. The object's nodes (locations) and class are set forth adjacent to each of the objects in the table.

OBJECT NAME	NODES	CLASS
Obdic	Host	Obdic
Sessions	Host	Session
Users	Host	User
Auctions	Host	Auction
Currencies	Host	Currency
Sale	Host	Sale

OBJECT NAME	NODES	CLASS
Sale Number	Host and Controller's workstation	Root
Bidlevel	Host, Currency Board Operator's Workstation, and Auctioneer's Display	Bidlevel
Currency board	Host, Television Display, and Saleroom Display	Root
Currentlot	Host, Controller's Workstation, Currency Board Operator's Workstation, Auctioneer's Display, Television Display, and Saleroom Display	Root
Bidding Flag	Host, Controller's Workstation, and Currency Board Operator's Workstation	Root
Ping-pong flag	Host and Controller's Workstation	Root
Number of bidders	Host, Controller's Workstation, and Auctioneer's Display	Root
Leading bidder	Host, Controller's Workstation, and Auctioneer's Display	Root

OBJECT NAME	NODES	CLASS
Last accepted bidder	Host, Controller's Workstation, Auctioneer's Display, Television Display, and Saleroom Display	Root
Biddisplay	Host and Controller's Workstation	Biddisplay
Logintab	Host	Logintab
Nextlots	Host and Controller's Workstation	Root
Bids	Host	Bid

Exemplary objects as set forth above are defined as follows:

"Obdic" is the object dictionary and contains the names of all objects, their locations, and a routing table which tells the computer the location of all objects and how to find each of the locations. This object is a basic part of the distributed object-based data processing system.

"Sessions" is a list of log-in times for each user, and where that user is located (e.g., Helsinki) for each session.

"Users" is a list of those entitled to use the system. These include paid subscribers to the system and the staff of the concern operating the system.

"Auctions" is information about the items to be auctioned, such as a list of the lots being sold.

"Currencies" is a list of the various currencies to be displayed and the exchange rates.

"Sale" is a set of details of the current sale which is in progress. The information in this object may be obtained from the "auctions" object.

"Sale number" is the number assigned to the current sale. A blank indicates that no auction is currently in progress.

"Bid level" is the amount of the current bid.

"Currency board" is a translation of the bid level into the various currencies contained in the "Currencies" object.

"Current lot" is the lot number and miscellaneous details of the current lot being sold.

"Bidding flag" is an indication whether bidding is in progress (i.e., the auction is not between two lots).

"Ping-pong flag" is an indication whether a ping-pong auction procedure is in progress.

"Number of bidders" is the number of bidders which bid through the system in the current round. This is obtained by instructing the computer to accumulate the number of bid signals received in any given round.

"Leading bidder" is the identification of the user whose bid signal was first received by the computer (including the first bidder in a ping-pong) or was "promoted" from the bid display by the controller.

"Last accepted bidder" is the identification of the bidder whose bid was accepted by the auctioneer in the last bidding round.

"Bid display" is a list of the bidders in the order in which the bids were received by the computer. This list is preferably limited in size to the top ten bidders.

"Log in table" is a table of users which have logged in for the current session.

"Next lots" is a description of several subsequent lots to be auctioned.

"Bids" is a table of bids and routines to process an incoming bid.

Exemplary classes for computer implementation of the televised auction system described herein are as follows:

"Session" class contains methods for determining the period a user is logged onto the system. This includes steps for determining log-in and log-out of a user.

"User" class contains methods which perform operations on the user file or another file such as a group file. These operations are, for example, adding, deleting, or updating a user (subscriber), and getting a user file for other operations.

"Currency" class contains methods for adding, deleting and changing currencies and currency exchange rates and for effecting exchange rate calculations.

"Auction" class contains methods for adding , deleting, and getting both an auction and a lot and for indicating when the hammer has fallen (as when an auction is terminated by acceptance of a final bid) and the next lot is to be auctioned.

"Date" class determines the current date and performs date manipulations.

"Sale" class contains methods to start an auction by initializing relevant objects to a useful state. For example, the bid display object must be set to zero or blank values, and the number of bidders object must be set to zero. These methods may be invoked by pressing a "start auction" button on the Controller's Workstation.

"Bid display" class contains methods to control the bid display, which is the display of the top ten bidders. These methods also update the bid display by adding or canceling a bid.

"Bid level" class contains methods to set a new bid level. For example, when the auctioneer accepts a bid, he states the price of the bid, and the currency operator provides an input specifying that level and invoking a method to update the bid level object.

"Bid" class contains a bid method, which updates the bid display object and the number of bidders object, and a bid cancel method, which removes bidders from the bid display,

replaces the canceled bidder, and updates the number of bidders.

"Login table" class maintains a list of the users, the log in table object, who have logged on the system.

In the preferred embodiment, the functions described above are performed by a general purpose computer, such that sold under the tradename "Stratus", and by personal computers, such as those using the Microsoft DOS operating system. The personal computers are programmed to advise the general purpose computer that they manage a copy of a particular object and are to be informed of changes to it (e.g., they will receive "set" requests informing them of a new value).

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a schematic diagram of the preferred hardware for implementation of an interactive, televised auction.

Figure 2 is an illustration of the auctioneer's display.

Figure 3 is an illustration of the controller's display.

Figure 4 is an illustration of a currency controller's display.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Figure 1 illustrates a combination of computers and communication elements which may be used to conduct a televised auction, or other interactive event. A host computer 2 is programmed with the distributed object-based transaction system described above and which has been tailored for a televised auction. The host computer 2 receives input from a subscriber by receiving signals generated at a subscriber terminal 4. The system is capable of receiving input from a large number of subscribers, but a single subscriber has been shown in the figures for illustration. In the ordinary arrangement, the subscribers are located at large distances from each other and from the location of the live auction and are, thus, large distances from the host computer. The preferred data connection between the subscriber terminal and the host computer is a telephone line, which is connected to a packet-switching network such as 6.

The subscriber also has a television 8 which receives broadcast signals by way of a satellite network 10, or other broadcast system. The screen of the subscriber television 10 is preferably divided into four areas. The first area 12 contains a number (the "paddle" number) identifying the bidder whose bid was accepted in the last round and the location of that bidder. This is preferably a display of the "last accepted bidder" object. Area 14 of the subscriber's television screen contains the amount of the last accepted bid in the selected currencies. This may be a

display of "currency board" object. Area 16 contains a video display of the article being auctioned, and area 18 contains a video display of the auctioneer.

The areas 12, 14, 16, and 18 are generated by a television mixer 20 which combines signals from the host computer which produce the displays of the currency board and last accepted bidder objects with video signals from one or more television cameras (not shown) in the auction room directed at the article being auctioned and the auctioneer. The display of objects generated by the host computer may be assembled by a television display generator 22 which supplies signals to the television mixer 20.

Several other display devices are provided for use by personnel associated with the auction. These are the auctioneer's display 24, the display on the currency converter workstation 26, and the display on the controllers's workstation 28. Each of these is connected to the host computer for supply by the host with signals representing the selected objects required by these articles. The preferred connection between the host and the workstations is by way of an X25 switch 30.

The auctioneer's display only receives signals from which it generates a display. An example of the auctioneer's display is shown in figure 2 wherein the number of the lot being sold is displayed at 32, the last accepted bid level at 34, the last accepted bidder at 36, and the number of bidders at 38. This display may be on a video screen located near the auctioneer such

that it is easily seen and is preferably projected on a semi-transparent screen located such that the auctioneer can view simultaneously both the bidders in the saleroom and the display.

The currency converter workstation 26 and the operator's workstation 28 are preferably personal computers capable of providing input to the system, including the host computer 2, relating to their functions in the conduct of the auction. In one embodiment, two personal computers are supplied with programs such that either may serve as the currency converter workstation or the operator's workstation. Alternatively, these workstations may be provided with programs exclusive to the selected function. The screens are preferably touch sensitive whereby touching a selected display feature invokes an appropriate method of the distributed object-based transaction system.

Figure 3 illustrates the display associated with the controller's workstation 28. This display contains the last accepted bidder at 40, the leading bidder at 42, the number of bidders at 44, the top ten bidders at 46, the number of the current sale at 48, and the lot number at 50. Touching one of the buttons 46 causes that bidder to be promoted to the leading bidder at 42, and touching the leading bidder display at 42 causes that bidder's bid to be accepted. Acceptance of a bid identifies that bidder as the "last accepted bidder".

A "ping-pong" bat 52 indicates that a ping-pong procedure is being conducted by the auctioneer, and a button 54 permits the

operator to remove the ping-pong symbol when the ping-pong is terminated. A "hammer fallen" button 56 allows the operator to indicate that the auctioneer has completed the sale of a lot, and touching this button causes the host computer to invoke the appropriate method to record such and to make any necessary updates of relevant objects.

Figure 4 illustrates the display on the currency board operator's workstation. This display shows the last accepted bid level at 58 and a series (in this example, ten) of pre-set increments above the last accepted bid at 60. Buttons 62 adjacent each of the increment indicators allow the increments themselves to be adjusted within the preset increment in case the auctioneer calls for a bid within the pre-set increments. A button 70 allows the currency board operator to set initial values. Touching this causes a keypad display to be superimposed on the display of figure 4 whereby the operator may key in the initial values. Box 72 displays the increment between the values shown in boxes 60, which in the illustration is £100. The current lot is displayed at 74.

Buttons 76 and 78 are provided for the case where the auctioneer calls for a price wholly outside the scale of the display. Button 76 causes the entire display to be shifted down by a preset amount, while button 78 causes the display to be shifted upward.

Button 80 allows the operator to exit from the display.

As the auction proceeds, the auctioneer states the price of the bid based *inter alia* upon the number of bidders. The currency operator enters this price by touching the button 60 having that price. The increment and number of prices displayed are designed to ensure that in the majority of cases, one of the areas 60 will show the price called for by the auctioneer. If the correct price is not displayed already, however, the currency operator uses the buttons 62, 76 or 78 to adjust the display until the correct price is displayed in one of the boxes 60. That price is then selected by the operator's touching that box, and this invokes methods to update the "last accepted bid" object with that value at all nodes.

The subscriber's terminal 4 allows the subscriber to interact with the auction being conducted in the saleroom and viewed on television 8. The terminal 4 may be of the type manufactured by Verifone and provides a slot for receiving an identification card (not shown) which has been provided to the subscriber by the operator of the system after appropriate credit investigations, or the like. When a subscriber wishes to participate in an auction, the card is "wiped" through the slot, and the terminal 4 reads the subscriber's information from the card. The subscriber is prompted by messages on display screen 66 to enter an identification number by way of key pad 68. The identification number is verified by an appropriate method in the host computer after which the subscriber is permitted to participate in the auction.

The subscriber's terminal includes buttons or other input devices for activation by the subscriber to allow the subscriber to signal the host computer that he wishes to bid or cancel a bid. For example, the subscriber's terminal has a "bid" button which signals the host computer that the subscriber has bid on the article being auctioned.

An auction utilizing the above described methods and devices would proceed as follows.

The controller would sign on at the controller's workstation, and the currency board operator would sign on at the currency board operator's workstation. The host computer then performs initializing methods which prepare the system to conduct the auction or perform system maintenance functions by updating any of the files, such as "user", "auctions", or the like.

A user begins by wiping the card issued by the auction operator in the slot on the subscriber's terminal and entering the assigned password. The terminal generates the "pin_login" method in the "sessions" class, and this method passes the user identification from the card and the password which has been entered by the user to the host for verification.

The televised auction is begun by the controller's pressing the "start auction" button on the controller's workstation when the auction in the saleroom is also begun. This activates the "begin sale" method which may be found in the "sale" class. The "next lot" method is called to set the "current lot" to "1", the number

of bids to zero, and the ping-pong flag to zero. The bid display is also initialized and the bidding flag is set to one to allow bids to be received.

When a remote bidder presses the "bid" button on the terminal 4, the host computer is requested to perform the "bid" method on the "bid" object which is this case is the identification of the bidder. The host also increases the "number of bidders" by one, sets the leading bidder by providing the identification of the bid to be received first, and updates the bid display. This process is followed for each subsequent bid.

When a bid is accepted by the auctioneer, the "leading bidder" is set and the identification of the bidder whose bid was accepted is moved to the "accepted bidder" location on the display such as at 42 in the display of figure 3. This is accomplished by the controller by his touching the proper one of the buttons 46, if the bidder is remote, or the button 43, if the accepted bidder is in the saleroom. The currency operator selects the proper button to display the "last accepted bid" level.

When the auctioneer begins a ping-pong auction, the "ping-pong" flag is set by the controller activating a "ping-pong" button and identifying the participants of the ping-pong. This causes the ping-pong symbol to be displayed on the various displays and permits only the participants of the ping-pong to be listed on the display as the last accepted bidder or the leading bidder. The identifications of the other bidders are placed on the bid display,

but none can be a "leading bidder" without the auctioneer first terminating the ping-pong.

When the auctioneer signals that the sale of the lot is completed, the "hammer fallen" method is invoked by the controller activating a "hammer fallen" button. This sets the bidding flag to zero, which indicates that no bids will be accepted by the host computer.

The steps followed by the distributed object-based transaction system are as follows. Activation of an input device, such as by touching a touch sensitive screen causes a "request" to be generated. That request comprises a method and an object and is in essence a statement to the computer to perform the stated method on the stated object. The computer first looks at the object table, such as one contained in the "obdic" object described above with respect to the interactive auction system. This table allows the computer to identify the devices on which the object resides, such as the host computer and one of the workstations. As noted above, it is a feature of the distributed object-based transaction system that the objects may reside on one or more separate devices. The computer determines the best route to the various locations of the object from the table and sends the instruction to all appropriate nodes where the object resides. The method is then performed on the object at all of the nodes on which the object resides. A reply is then sent if the selected method or original request called for a reply.

It will be appreciated that a unique transaction system has been described. Modifications within the scope of the appended claims will be apparent to those of skill in the art.

We Claim:

1. An interactive system comprising:

host computer means for performing data operations;

a plurality of workstation terminal means connected to said host computer for displaying data from said host computer and for providing input to said host computer,

television network means for transmitting video information to a plurality of subscriber video terminals; and

a plurality of subscriber data terminal means for supplying data to said host computer.

2. An interactive network according to claim 1 further comprising mixing means for supplying said data from said host computer to said television network for combination with said video information.

3. An interactive system according to claim 2 wherein said television network comprises television signal broadcast means for supplying said video information to said subscriber video terminals and said subscriber terminal means comprises telephone transmission means for supplying said data from said subscriber terminal means to said host computer.

4. An interactive system according to claim 3 wherein said video information produces an image of an item being sold on each of said subscriber video terminals and said data contains information about the price of said item.

5. An interactive system according to claim 4 wherein at least one of said workstation terminal means displays said information about the price of said item.

6. An interactive system according to claim 5 wherein said information about the price of said item comprises the last bid for said item which has been accepted in an auction.

7. A method for conducting a transaction comprising

providing a host computer with data regarding an item,

providing an input terminal to at least one subscriber for transmitting signals to said host computer,

providing a video terminal for displaying an image of said item to said subscriber,

wherein said signals indicate transaction information with respect to said item.

8. A transaction system comprising a host computer for performing data operations, a subscriber terminal for transmitting data from a subscriber to said host computer, video means for transmitting an image of an item to said subscriber, and a workstation terminal for displaying data from said host computer and for transmitting data to said host computer.

9. A method for processing data comprising

providing a plurality of computing means,

defining a plurality of objects by allocating memory space in said computing means for each of said objects and by naming each of said objects,

providing at least one method for performing an operation with respect to said objects,

wherein said memory space for at least one of said objects is allocated in a plurality of said computing means and said method includes the step of updating the value of said object at all memory spaces assigned to said object upon completion of said method.

10. A method according to claim 9 wherein said objects are related to an auction, and said at least one method comprises a plurality of methods relating to an auction.

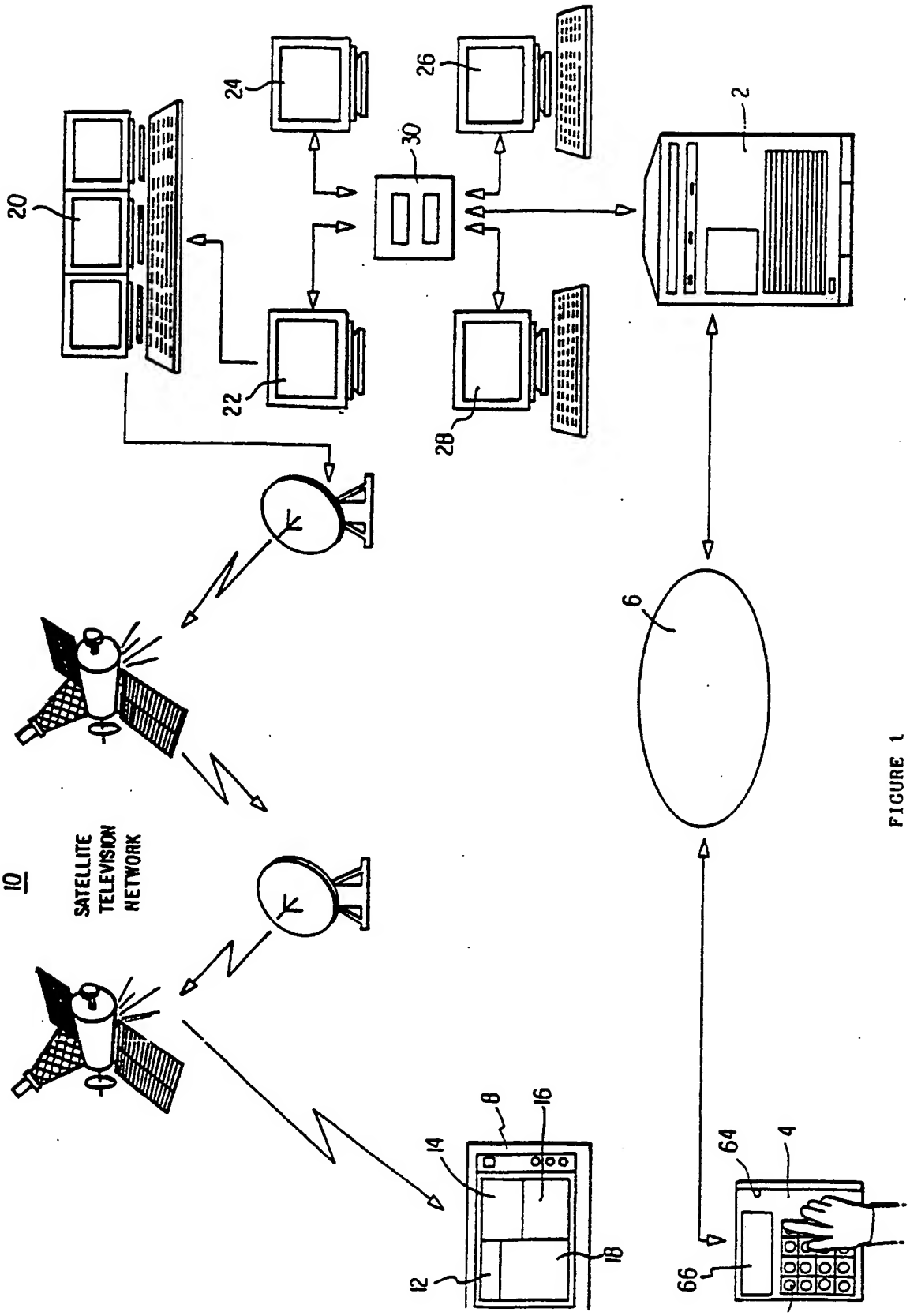

10
SATELLITE TELEVISION NETWORK
2
4
6
8
12
14
16
18
20
22
24
26
28
30
64
66

FIGURE 1

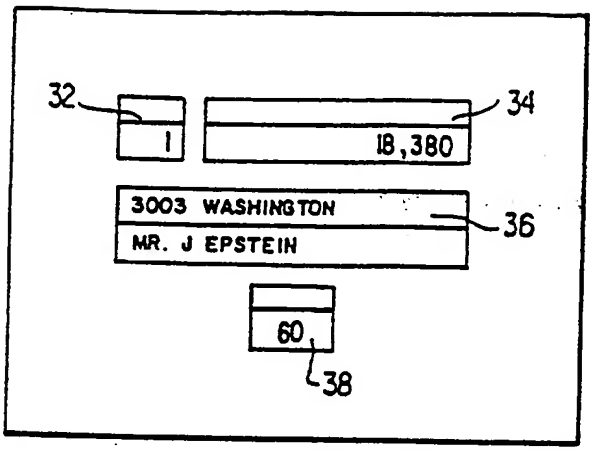
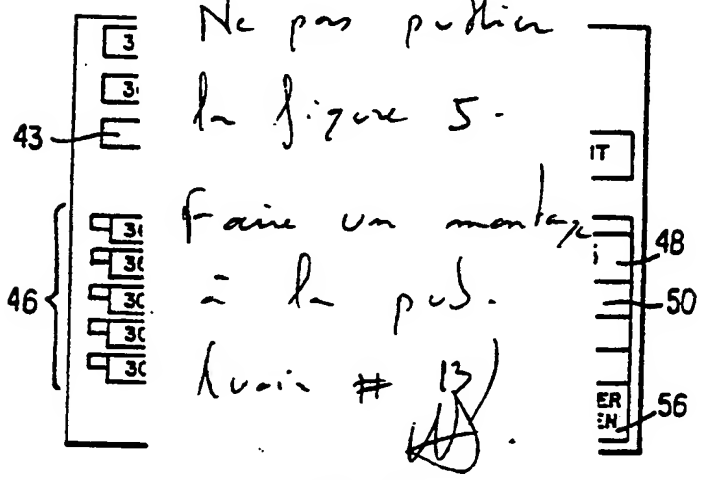


FIG. 2



Ne pas publier la figure 5. Faire un montage à la pub. Avoir # 13

FIG. 3

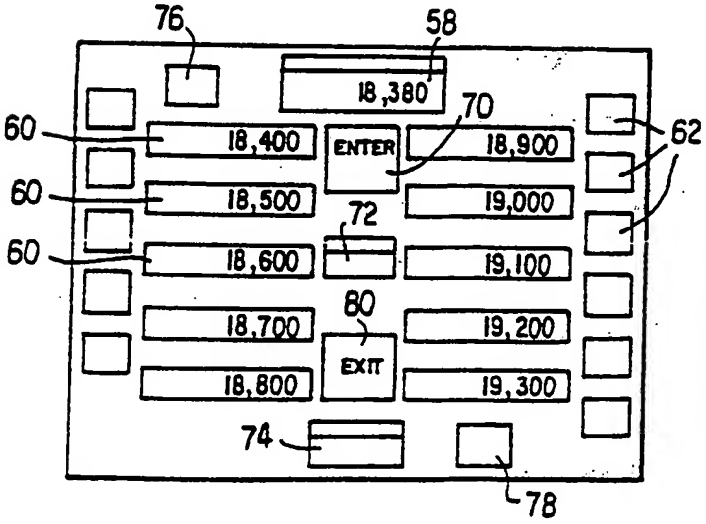


FIG. 4

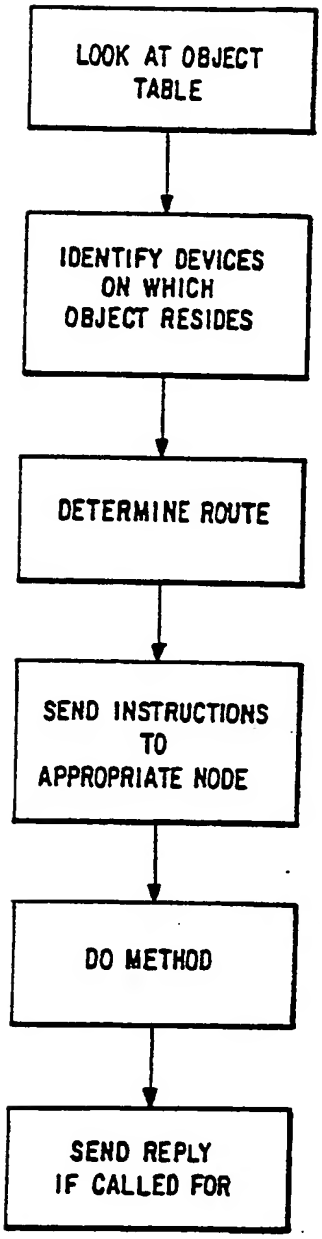


FIG. 5

Not to be taken into consideration for the international procedure.

INTERNATIONAL SEARCH REPORT

International Application No PCT/GB 92/00337

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all)[6]

According to International Patent Classification (IPC) or to both National Classification and IPC

Int.Cl. 5 H04N7/173

II. FIELDS SEARCHED

Minimum Documentation Searched[7]

Classification System	Classification Symbols
Int.Cl. 5	H04N ; H04Q

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched[8]

III. DOCUMENTS CONSIDERED TO BE RELEVANT[9]

Category *	Citation of Document,[11] with indication, where appropriate, of the relevant passages[12]	Relevant to Claim No.[13]
Y A	EP,A,0 342 295 (KATZ) 23 November 1989 see the whole document	1,7,8,9 2-6,10
Y	FUNKSCHAU vol. 8, no. 19, September 1989, MÜNCHEN pages 46 - 51; V.LUGGER: 'NEUES KONZEPT VERBINDET TV UND TELEFON'	1,7,8,9
A	see the whole document	2-5
A	16TH INTERNATIONAL TV SYMPOSIUM 16-22 JUNE 1989 MONTREUX, SWITZERLAND pages 554 - 559; S.ROBERTS ET AL.: 'VIDEOTEX AND INTERACTIVE SERVICES ON TREE&BRANCH CABLE TV NETWORKS' see page 557, line 6 - page 558, line 37	1-5,7-9

-/--

* Special categories of cited documents :[10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report
06 MAY 1992	27.05.92
International Searching Authority	**Signature of Authorized Officer**
EUROPEAN PATENT OFFICE	GREVE M.P.

III. DOCUMENTS CONSIDERED TO BE RELEVANT (CONTINUED FROM THE SECOND SHEET)

Category °	Citation of Document, with indication, where appropriate, of the relevant passages	Relevant to Claim No.
A	GB,A,2 194 088 (AXON RESEARCH LIMITED) 24 February 1988 see page 4, line 41 - page 5, line 40 ---	1-5,7-9
A	DE,A,3 316 804 (WESTERN ELECTRIC CO.) 10 November 1983 see the whole document ---	1-5,7-9

ANNEX TO THE INTERNATIONAL SEARCH REPORT ON INTERNATIONAL PATENT APPLICATION NO. GB 9200337 SA 56781

This annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report. The members are as contained in the European Patent Office EDP file on 06/05/92
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0342295	23-11-89	US-A- 4845739	04-07-89
		US-A- 5073929	17-12-91
		US-A- 5014298	07-05-91
GB-A-2194088	24-02-88	None	
DE-A-3316804	10-11-83	GB-A- 2120507	30-11-83
		JP-A- 58206273	01-12-83